```
Message-ID: <3919F38B.1B780FE3@lucent.com>
Date: Wed, 10 May 2000 18:40:59 -0500
From: clarisse@lucent.com
Reply-To: clarisse@lucent.com
Organization: Lucent Technoliges
X-Mailer: Mozilla 4.61 [en] (Win98; U)
X-Accept-Language: en
MIME-Version: 1.0
To: "Dunn, J P (Jim)" <jdunn@lucent.com>,
        "Collet, Pascal" <pcollet@lucent.com>,
        "Westergren, Bruce A" <westergren@lucent.com>
Subject: USE THIS ONE: BTJ CCBNS smaller version
Content-Type: multipart/mixed;
 boundary="------------0AA1A1A2828E04F8C7AE40D7"
X-Status: $$$$
X-UID: 0000001225


This is a multi-part message in MIME format.
--------------0AA1A1A2828E04F8C7AE40D7
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit

It turns out the previous version I just sent was over 5MB probably
because one of the pictures was inserted directly from PPT, I converted
to gif and reinserted it in this new version. Should save much space...

--------------0AA1A1A2828E04F8C7AE40D7
Content-Type: application/msword;
 name="1_BTL_ccbns_2.doc"
Content-Transfer-Encoding: base64
Content-Disposition: inline;
 filename="1_BTL_ccbns_2.doc"
```

```
Message-ID: <39355086.FFF54028@lucent.com>
Date: Wed, 31 May 2000 12:48:55 -0500
From: olivier clarisse <clarisse@lucent.com>
Organization: sas
X-Mailer: Mozilla 4.7 [en] (X11; U; Linux 2.2.12-20 i586)
X-Accept-Language: en
MIME-Version: 1.0
To: "Colbert, Raymond O" <rcolbert@lucent.com>,
        "Westenkirchner, Paul M" <westenkirchner@lucent.com>
CC: "Collet, Pascal" <pcollet@lucent.com>,
        "Velazquez, Lizette" <lizette@lucent.com>,
        "Nanke, Trent R" <tnanke@lucent.com>,
        "Westergren, Bruce A" <westergren@lucent.com>,
        "Dunn, J P (Jim)" <jdunn@lucent.com>
Subject: [Fwd: FullCircle conference debriefing]
Content-Type: multipart/mixed;
  boundary="------------51F625F3F76249698D771971"
X-Status: $$$$
X-UID: 0000001249

This is a multi-part message in MIME format.
--------------51F625F3F76249698D771971
Content-Type: text/plain; charset=iso-8859-2
Content-Transfer-Encoding: 7bit
```

Ray and Paull,

I take this opportunity to share impressions and results
from the Full Circle conference. We can meet soon
and get into more details to plan the next opportunities
for collaboration where appropriate.

I am forwarding an exciting report that Pascal wrote
as soon as he returned. We all share a similar experienced
from the interest and questions we received from
conference participants. While the demo sessions
were limited per the schedule, it turns out we were
constantly demonstrating and discussing with attendants
throughout all 2.5 the conference days.

In one such instance, Bruce was working on the LSS code
after hours merging call features, and he ended up
giving a private demonstration to a delegation
of Asian customers (Excell customers) who were very
glad to experience VoIP and conferencing
between parties live on top of Lucent Softswitch.

The demonstration system (and demo MMRS system)
worked flawlessly throughout the 2.5 days.
[With the exception is a 3com hub that had to be
replaced after a customer reported packet loses
on several of the Webphones.]

All Webphones were provisionned with 4 special feature
buttons to setup and join 3 way and 6 way conferencing
sessions directly using the MMRS. I used this feature
demonstrate one click to multiway conferencing on MMRS.

Portal Services:

An Evolution of Voice Features

Authors

Olivier B. Clarisse

Pascal Collet

J. P. (Jim) Dunn

Bruce A. Westergren

Broadband Access and the Evolution of Voice Features

Abstract

Broadband Access will not only change the character of data transmission and user platform, it will have a profound effect on voice features. In the converged network the feature developer has new tools to control configuration, activation, and content delivery to the end user. These new resources include web pages, electronic forms, interactive graphics, bookmarks, and JAVA applets.

The Broadband Network Services Project, sponsored by Lucent Technologies Switching Systems Group, has established a demonstration environment that shows how to develop these new interactive features. The goal is to establish frameworks that will speed feature development, demonstrate the feasibility of Internet enabled telephony services in an open-source environment, and to integrate these services with video and high speed internet access in a broadband endpoint. This paper discusses the demonstration environment, the new user platform hardware and software, the roles of converged data, and programmable switches. Also, this paper presents an overview of one of the initial features.

Broadband Access and the Evolution of Voice Features

Overview

One of the goals of the Broadband Network Services Project (BNSP) is to demonstrate the feasibility of Internet enabled telephony services in an open-source environment with very limited development time and resources. Ultimately the goal is to integrate these voice services with broadband video and high speed internet access in a broadband endpoint. In a converged Internet and Telephony network a number of trends are emerging that allow the delivery of innovative services. One of these trends is the development of open Application Program Interfaces (API) that comes from the Computer Telephony Industry (CTI). Another is the Web based servers that support data sharing, forms, and hyperlinks. While a third is the personal computer / network appliance trend that enables Internet protocols and browsers on a variety of endpoints. In the first phase the BNSP draws upon all of these capabilities to demonstrate features in what we call an Internet Telephony enabled portal service that makes any user a high powered expert. Portals are implemented as a combination of web pages that are either written in HTML or generated via Java Servlets, Common Graphic Interface (CGI), Java Applets, or XML. The personal preferences are maintained in a database implemented in SQL, LDAP, and other database technologies. Access to the portals is via a browser using TCP/IP protocol. Services are

Broadband Access and the Evolution of Voice Features

presented as hyperlinks that require one click to launch complex software routines that are totally unseen by the user. Again these routines may be implemented using Java Servlets, CGI, ActiveX, or other control mechanism to run C programs, java programs or deliver java scripts and applets. Future work will integrate the internet voice services with the Broadband Fiber Access[1] technology.

Demonstration Network

Although the initial demonstration network is implemented on an addressable LAN, the same network nodes and services will function on any corporate WAN, wireless LAN, Intranet, Internet, or Broadband Fiber Access (BFA) access network. Figure 1 shows the network configuration for the initial demonstration services.

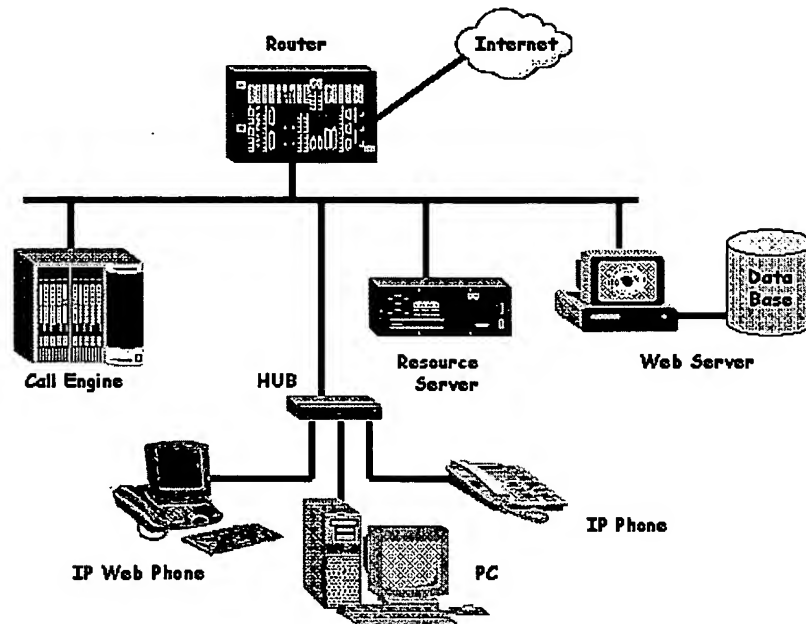# Broadband Access and the Evolution of Voice Features

Figure 1: Demonstration Network

The router provides access to the internet and the ability to address messages between servers and endpoints. The hub provides additional port capacity to connect a number of endpoints. Figure 1 shows the three types of endpoints used in the demonstration:

- IP Web Phones - equipped with a 10 base T network interface, a Mantra protocol handler, and software browser

- PC - equipped with a 10 base T network interface and Microsoft Netmeeting

- IP Phones - equipped with a 10 base T network interface and an ITU
  H.323 protocol handler

The Ethernet network provides standard TCP/IP services with no special

functions or extensions.

Call Engine

The call engine for the BNSP is Lucent SoftSwitch from Bell Labs Research

(LSS)[2], which provides a number of capabilities including the Call

Coordinator (CC) and the Device Servers. All of these elements are

described below.

Call Coordinator

The Call Coordinator[3] provides the connection control for signaling

information and arranges the media paths between device servers and

endpoints. The underpinnings of the CC is implemented by a distributed call

model called Mantra. The CC has the following responsibilities:

- maintains the state of all calls

- coordinates communications between device servers and

resource servers

- implements a call back API for communicating with the

Service Provider Servlet and User Feature Applets.

- for each call it maintains a hierarchical namespace using

the STYX protocol for remote communications

- provides for routing of calls based on a directory service

dial plan strategy. The dial plan is implemented using simple

flat files that associates a phone number with and endpoint IP

address.

Device Servers

The LSS implementation includes support for device servers which perform

the traditional functions of a gateway and optionally a gatekeeper. The

gateway functionality is the translation between the endpoint protocol and

the mantra protocol. In the current implementation we are using an ITU

H.323 protocol device server. The gatekeeper functionality allows an ITU

H.323 protocol endpoint to register with the device server and subsequently

inform the CC that a new endpoint is available. The commands used to

communicate with the CC are *addBox* and *dropBox*. We are currently using

Netmeeting and Sagitta IP phones as our ITU H.323 protocol endpoints.

An additional IP web phone endpoint is also used that implements the

Mantra protocol and therefore does not require an external device server

resource. It contains the necessary code to interface with the CC and perform the registration functions of a gatekeeper.

Resource Servers

The demonstration uses both a hardware based resource server and a software based resource server. These elements offer media resources to the endpoints such as tone detection, playing announcements, recording announcements and general speech services. The software Resource server implementation we used is a version of the Elemedia Programmable Media Server (EPMS). This engine can run on the same physical hardware as the call controller or for performance reasons on a remote machine. The EPMS is a media processing engine capable of terminating RTP/UDP/IP or AAL-1/AA1-2 audio streams and performing some processing on those streams. Additional capabilities exist to provide conferencing through audio stream mixing.

As an alternative to the EPMS software resource server we have integrated the Multi-Media Resource Server (MMRS) from the 7R/E project. This is a separate piece of hardware and software configured to support DTMF tone detection, stored announcement playback,

Broadband Access and the Evolution of Voice Features

automatic speech recognition, text to speech, voice dialing, multimedia bridging, internet call waiting host and voice mail hosting. The target for using this platform is for high performance bridging of voice streams in our conferencing and broadband applications.

The communications protocol used to communicate between the resource servers and the CC is MGCP[4].

End Points

Multimedia end-points are the ultimate delivery agents of the service portal experience to the end-users. End-points enable services to reach customers. With the sophisticated protocol and user interfaces enabled by the Internet, a popular end-point can reach millions of customers driving the need for new services and ultimately driving the required changes to the network infrastructure to support the new service and access demand.

If it were not for the availability of free and sophisticated web browsers in the early 1990s, the ultimate Internet experience might have been confined to research labs for another decade. Similarly Internet Telephony end-points could be leveraged to drive a new service infrastructure and the demand for ubiquitous broadband connections.

Broadband Access and the Evolution of Voice Features

There is a growing range of devices available (or soon to be) that can provide a "combined" voice and Internet experience to users everywhere including:

-       Cellular phones for Internet access.

-       Wired IP phones: business phones providing Voice over IP (VoIP) using with a direct Ethernet link connection. For LAN, DSL, cable or Wavelan access.

-       IP Web phones: IP phones with a touch screen enabling simultaneous email, browsing and VoIP calls.

-       Play stations (possibly a future IP end-point).

-       TV and set-top-boxes.

-       Audio capable pocket organizers and palm devices (wireless or wired).

-       Netmeeting enabled notebooks, portables, personal computers or workstations.

These devices are widely known (except for the more recent IP based phones.) The BNSP team has been using "standard" IP phones as well as custom IP Web phones to experience the portal services as end-users and to demonstrate the types of services best rendered on each of the devices. As a result, the service portal experience is almost identical between a personal

Broadband Access and the Evolution of Voice Features

computer end-point and a custom IP Web phone, while service experience is limited to voice and voice announcements and menus using a "standard" IP phone (Figure 2).

Figure 2: Business IP Phone

"Standard" IP phones provide a typical 24 to 32 keys business phone, a handset, and a two line display. The phones are ready to connect to an Ethernet wall connection. IP phones available provide VoIP based on the ITU H.323 protocol or SIP protocols for example. These IP phones are limited by a 2 line display preventing access to the web for browsing and data communication.

A Web phone is a cross between a single-board computer and a traditional telephone (an embedded computer mother-board with digital audio support, and a touch-sensitive screen). Some of the first web phones became available in Fall 1998 and in 1999, including the Philips IS2630 (Figure 3)

Broadband Access and the Evolution of Voice Features

and the Alcatel Webtouch phones. Such devices combine a complete analog phone device and a diskless processor providing email, browser and a variety of organizer applications (call list and directory manager, notepad, calendar, etc.). Web phones are limited by the speed of dial-up modems, prohibiting VoIP.
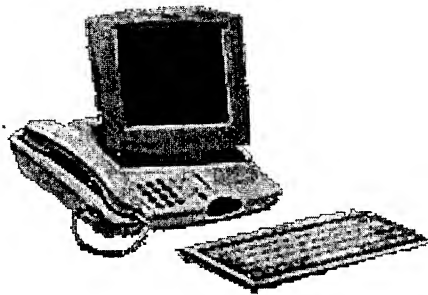
Figure 3: IS2630 phone

Custom IP Web phones

A 1998 experiment at Bell Labs research project led by Venkatesh Krishnaswamy provided the first demonstration prototypes of IP Web phones based on modified IS2630 Web phones. These IP Web phones enabled VoIP calls using the LSS. We reused the results of this experiment and included our enhanced version of IS2630 Web phones as part of our IP end-point infrastructure.

The IP Web phones are unlike today's personal computers, they are connected all the time, are always available to receive and send data. They

Broadband Access and the Evolution of Voice Features

restart (or powered up) and become operational in less than 10 Seconds, are ready for emergency calls, appear as robust and ubiquitous as POTS telephone. Yet the IS2630 IP Web phones use a StrongArm 1100 processor (Intel), 16 Mbytes of memory, 4 Mbytes of flash memory (for stored applications and data), a light weight custom Operating System (The Inferno OS on the IS2630) and an application suite providing: VoIP, browsing, email, software video rendering, and phone top applications.

IP Web Phone Customization

The original IS2630 phones received the following changes to enable IP data and voice services:

- External hardware addition to reuse the existing DSP/Voice CODEC (apply it to capture and render packet audio) and redirect the resulting audio I/O to the existing analog voice system. This enables handset audio, and speaker phone (as well as existing volume control features) to work identically with packet based audio or analog audio.

- Extend PCMCIA interface to support 5V (as well as the original 3.3V) enabling the use of a broader range of Ethernet access card.

- Software changes include: audio driver (RTP/RTCP), media manager, telephony server, configuration manager (remote

Broadband Access and the Evolution of Voice Features

LSS provisioning), changed DSP driver, upgraded IP stack and selected Ethernet card driver.

The IP Web phone experience is greatly enhanced over the original Web phone by the speed of direct Internet access, making this a very attractive device for service delivery to the business office via LAN or Wire LAN, or to the home via DSL, cable modem or high speed wireless.

For example, using a 10Base-T network interface card, an IP Web phone enables a quality audio call over IP to proceed unchallenged while very fast online information access is available (e.g. from a browser and while accessing email). In addition the phone functionality and software are seamlessly extensible to the network were servers can provide complimentary applications on demand.

The next generation of IP Web phones could be enabled by a new IP phone platform provided by Lucent BCS. The new hardware based on Lucent ME "IP Phone on a chip" technology, uses USB (bus), enabling platform evolution towards video extensions, video conferencing device and the realization of broadband IP Web phones. Other proposed designs include the Intel's new Linux based Web phone platform.

Broadband Access and the Evolution of Voice Features

Web Server Applications

The demonstration applications for the BNSP have been implemented as a portal service (Figure 4). Just as corporations use portals[5] as a new way to bring together employees, customers and suppliers (whether or not they share the same infrastructures), the service portal benefits telephony users by seamlessly providing capabilities of LSS to different clients such as computers and webphones. From any type of web-enabled clients, users will be able to read their mail, participate in multi-site/companies conferences, access/share documents, etc from anywhere (commuter, at home, from abroad, from customer sites, etc.).

Broadband Access and the Evolution of Voice Features
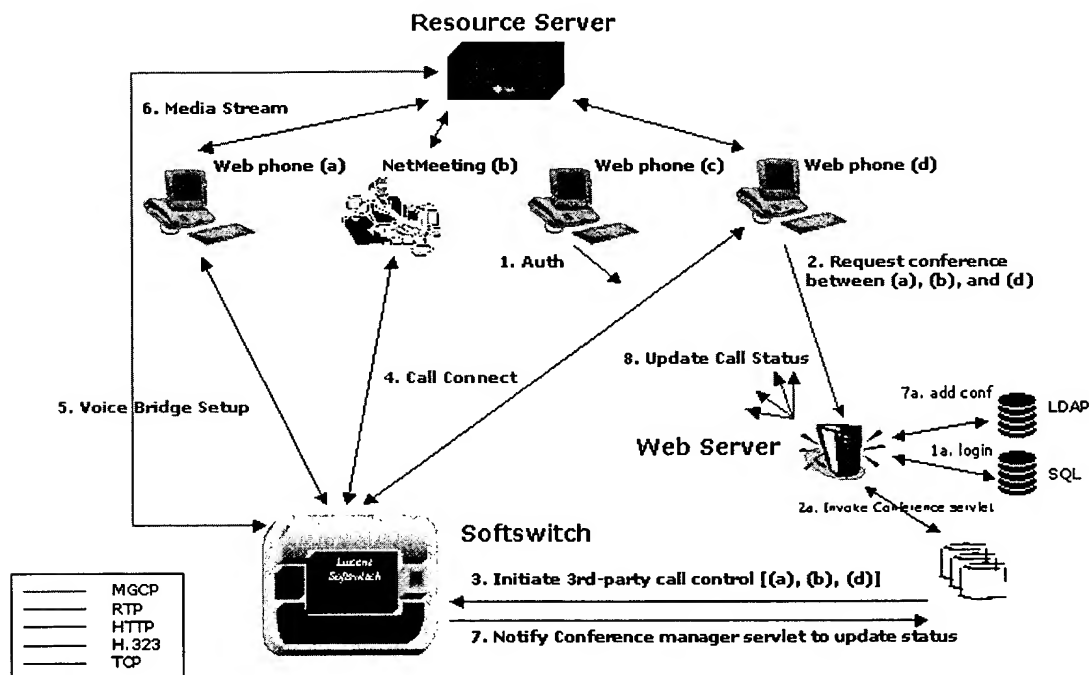

Figure 4: Portal Service Scenario

In order to deploy portals (corporate or service), whatever final applications are intended, one needs to put in place several common services [6]

- Membership services for establishing a portal community,

- Presentation services for creating/maintaining page layout,

- Personalization services for allowing users to create their own virtual office space,

- Security services for remote access account or extranet via user authentication,

Broadband Access and the Evolution of Voice Features

- Integration services for bringing together applications using HTML and XML.

We have developed our own portal services that support a minimal set of the common services described earlier. Although all services are provided via an open source Linux machine, these services could have been implemented on other proprietary operating systems. The Personalization service is provided via an SQL server that let user register themselves, change their account information, and so on. The Membership service is provided via LDAP servers, one for the directory that is compatible with NetMeeting users, and the other one for dynamically deploying services. The directory allows users to browse online users and setup conferences between multi-users in a very friendly manner. All call controls are web-based allowing one to add, drop, and setup conferences with the click of a mouse. The open nature of the LSS API is fully used in that context as it allowed us to control and operate call control remotely via Java Servlets running from our portal server distinct from the LSS server.

Finally, a telephony service creation tool is provided to the user to manage its own services. It is based on the following simple query that is general enough to permit the deployment of complex services for a user perspective.

query: "FROM <a user> WHEN <a date> WHERE <a location>

WITH <a pwd>

USING <an application>"

result: <phone number> | <announcement number>

some examples:

1) FROM my_boss WHEN office_hours WHERE office

   -> my_office_phone_number

2) FROM my_mom

-> busy_tone

The user's services are stored in a SQL database via the Personalization and Membership services. The user has access to its services all the time and can create, delete, modify, activate or deactivate services in a very simple manner. The activation/deactivation can be based on manual operation or automatic procedure based on an expiration/alarm trigger.

Service Example

Web Portal Design

We have integrated on a Linux box an Apache web server, University of Michigan LDAP server, and a MySQL server for our architecture. The web

server allows users to have simple access to their information (account, address book, call records, buddy list, etc.) and to carry out complex telephony operations seamlessly. The next-generation telephony services such as CLICK2CONF will integrate different technologies to provide new level of simplicity combined with enhanced features to user. In this way, in the context of our lab, we can setup a conference with the click of a mouse (or touching the screen), decide later on to add an expert in the same way, and at any time during the conference call chat privately with any participants to increase the effectiveness of the conference call. As of today, all of these would be quite tedious to setup. The web is the perfect support to provide these kind of services to any type of browser-enabled multimedia endpoints such as desktop computers, IP Web Phone, Internet appliances, etc.

Web Pages

Figure 5 shows the home page for the conference portal developed for the demonstration.

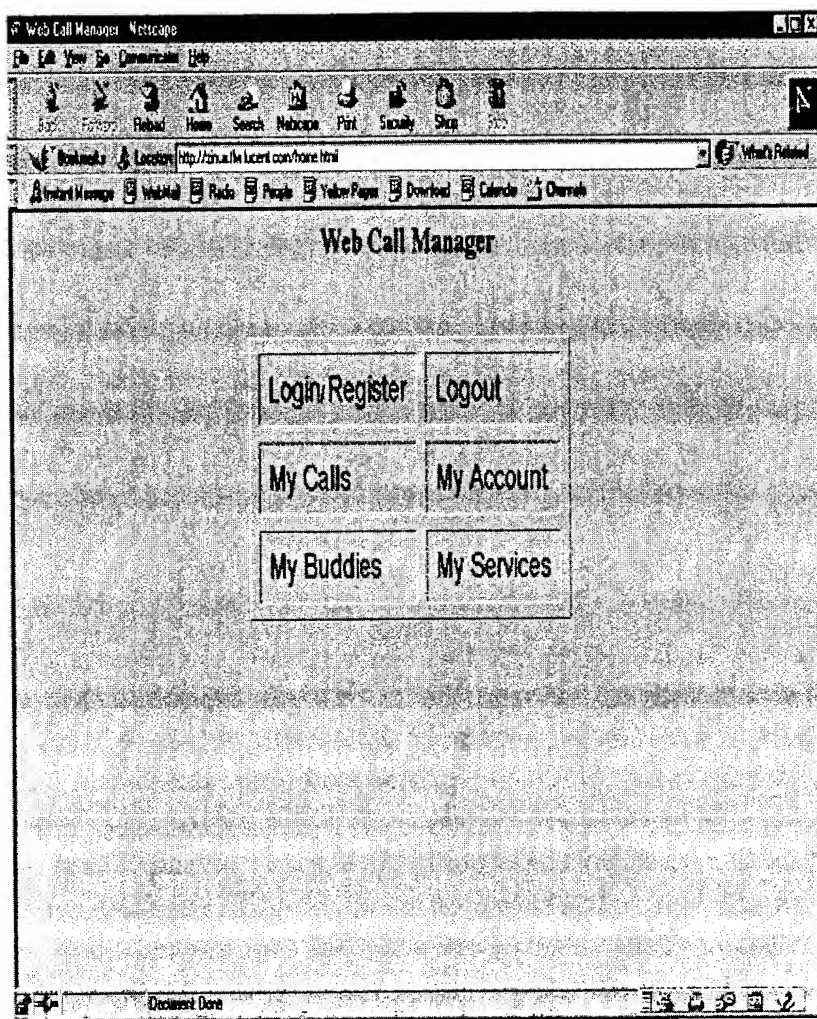# Broadband Access and the Evolution of Voice Features

Figure 5: Conference Service Portal Home Page

The web call manager, which is described in more detail later, is the controlling user interface for the demonstration. The page can be loaded on a PC, IP Web Phone or any network device with a web browser.

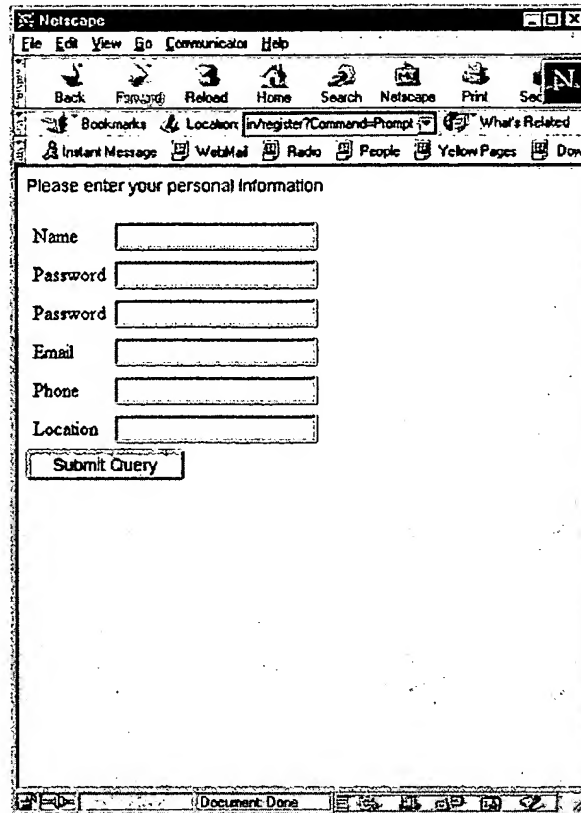# Broadband Access and the Evolution of Voice Features

Figure 6: Registration Page

The registration page (See Figure 6) allows the user to register with the SQL database and provide the phone number and IP address information to the service.

Figure 7 shows the conference screen. This screen allows the user to select among registered users to establish a conference call. Any user can establish a call, drop, or add themselves or another user from the call at anytime. The web call manager translates the selections into the required commands to the LSS which establishes the media streams between each

Broadband Access and the Evolution of Voice Features

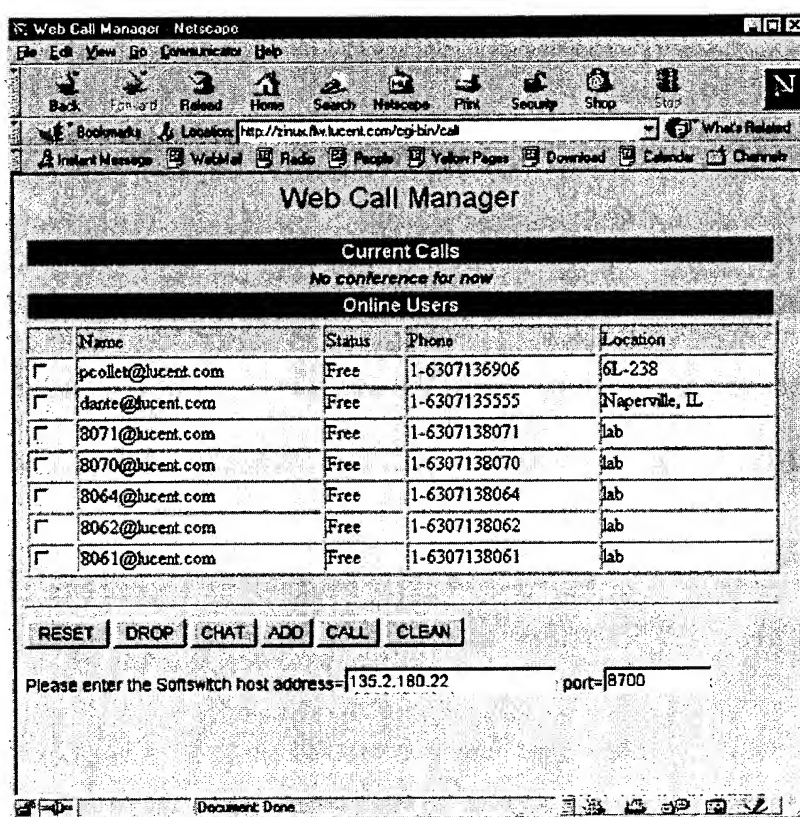endpoint and the conference bridge. This process is described in more detail below.

Figure 7: Conference Screen

Web Call Manager

The Web Call Manager is a family of Java Servlets dynamically generating HTML contents that supports all the features we have mentioned earlier. Special attention has been paid to the HTML pages so they can be quickly

accessed by any type of clients (no plug-in, or Java applet necessary). This model tends to load the server more but provides the ubiquity service providers are looking for: easy to install, easy to manage, and no special requirements for the clients.

Databases

The LDAP[7] server acts as our user directory and conference repository database. All users have access via the Web Call Manager to online users that previously logged into the system. The mobility aspect of our demonstration is thus embedded into the LDAP server. Wherever a user logged in, that client will become its phone where all ingress calls will be routed. When the user is logged off, the voicemail will take the call. All the user has to do is log in where they want to be reached.

The LDAP server is also used to keep track of current conferences with its participants. At any time, the Web Call Manager gives the logged in user the ability to check who can participate in the conference, add any logged in user, or drop if necessary. In coordination with the CC, the conference object is kept up-to-date as users join or leave the conference. The conference repository is always able to reflect the current situation. NetMeeting users also have the possibility to use this LDAP server as their directory service so the Web Call Manager community can reach them

directly at their desk. In addition, the LDAP server serves as the buddy list repository for users so they can keep an eye on who is online, where, and whether or not they can be reached in real-time.

A SQL server stores user account information (name, password, email address, and preferences) and routing services.

We want to be able to leverage cross-platform development benefits since we have in our lab quite a few of different types of endpoints.

Lucent SoftSwitch

The programming language used in the LSS implementation is JAVA and its virtual machine. JAVA is object oriented providing the capability to encapsulate standard call processing functionality and allow these features to be overridden at the application level. This provides for the ability to implement new services by users other than the internal system developers. The concept exposed for the third party programmability is the Service Provider Servlet (SPS).

Service Programming Interface

To program new service offerings using the LSS we use the SPS. This uses a call back API to intercept the triggers in the call model. The SPS initially

Broadband Access and the Evolution of Voice Features

reads in a configuration database to setup the device groups, protocol

handlers, and call capacity data. It is then ready to begin processing calls.

A sample call flow would use the following call backs:

- onInitCaller

- onInitCallee

- onCalleeRouteSelect

- onCallProceeding

- onCallAlert

- onCallConnect

- onCallBusy

- onCallDisconnect

At each point in the call flow the SPS can choose to allow normal processing

or intervene to add call features. Among these features are the ability to

invoke user feature applets. These provide service control on a user by user

basis. They may run on the same processor as the SPS or remotely.

Our sample service encapsulates all call logic in the SPS. There are two

control strategies provided in the implemented SPS for processing calls. The

Broadband Access and the Evolution of Voice Features

normal first party call control is handled when the endpoint initiates a call. The CC receives a CallRequest

from a device server and begins the call processing flow. The other method is through the third party call control initiated from the Web Call Manager. A simple protocol has been implemented to accept call requests from the browser and through the SPS to initiate processing of users in a call. Upon acceptance a callid is returned so additional users may be added or current users can be dropped. The SPS implements internal state information on the status of the calls which allows the service to distinguish the difference of third party vs. user first party call control. During the processing of the call user announcements are presented to the user upon connect using stored announcements from a resource server. The command protocol used to control adding and dropping endpoints from a call:

- o COMMAND=CALL&PHONE=16307138070&PHONE=1630 7138071
- o COMMAND=ADD&CALLID=0&PHONE=16307138059
- o COMMAND=DROP&CALLID=0&PHONE=16307138070

Conclusion

<< insert conclusion paragraph here >>

Broadband Access and the Evolution of Voice Features

References

1. D. Bruce buchholz, et.al., Broadband Fiber Access: A Fiber -to -the-Customer Access Architecture, , Bell Labs Technical Journal January - March 1999.

2. Ramnath A. Lakshmi-Ratan, The Lucent Technologies LSS - Realizing the Promise of Convergence, Bell Labs Technical Journal April-June 1999.

3. Henry Peter, Towards Unifying Distributed and Diverse Resources, Internal Document, February 14, 1999.

4. MGCP - Internet Engineering Task Force "Media Gateway Control Protocol," IETF Drafts, April-July 1999, http://www.ietf.org/ids.by.wg/megaco.html

5. http://www.opendesk.com

6. Mike Koller, "Sun-Netscape Alliance Unveils Portal Package", Internet Week, 2/28/2000.

Broadband Access and the Evolution of Voice Features

7. M. Wahl, T. Howes, S. Kille, "Lightweight Directory Access Protocol (v3)", RFC2251, Dec. 1997.

Glossary

7R/E	7 R/Evolution
AAL	ATM Adaptation Layer
API	Application Program Interfaces
ATM	Asynchronous Transfer Mode
BCS	Business Communications Services
BNSP	Broadband Network Services Project
CC	Call Coordinator
CGI	Common Graphics Interface
CODEC	COder DECoder
CPE	Customer Premise Equipment
CTI	Computer Telephony Industry
DSL	Digital Subscriber Loop
DSP	Digital Signal Processor
DTMF	Dual Tone Multi Frequency
HTML	Hyper Text Markup Language
IP	Internet Protocol
I/O	Input / Output
ITU	International Telecommunications Union
LDAP	Lazy Data Access Protocol
LSS	Lucent SoftSwitch from Bell Labs Research
ME	Micro Electronics
MMRS	Multi-Media Resource Server
PC	Personal Computer
PCMCIA	Personal Computer Memory Control Interface Adapter
EPMS	Elemedia Programmable Media Server
POTS	Plane Old Telephone Service
RTP	Real Time Protocol
RTCP	Real Time Control Protocol
SIP	Something I Protocol
SPS	Service Provider Servlet
SQL	Structured Query Language
UDP	Universal Data Protocol

Broadband Access and the Evolution of Voice Features

VoIP   Voice over IP

XML   Extensible Markup Language

THIS PAGE BLANK (USPTO)

THIS PAGE BLANK (USPTO)